The 6th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI 2009)

Jerk-Bounded Trajectory Generation Method Using Digital Convolution

Geon Lee and Youngjin Choi*
Hanyang University
{emerald,cyj}@hanyang.ac.kr

Jinhyun Kim
Seoul National Univ. of Tech.
jinhyun@snut.ac.kr

Suk-Joong Kim
ED Co.
sjkim@ed.co.kr

Abstract − This paper proposes a jerk-bounded trajectory generation method using a digital convolution. The suggested jerk-bounded trajectory is able to reduce unexpected damages for a robot motion control system and to improve a tracking accuracy/speed of the control system. Also, it can be implemented in real-time because it requires low computational loads. The effectiveness of the suggested method is shown through numerical simulations for a point-to-point (PTP) motion generation application.

Keywords − Trajectory generation, Jerk-bounded, Digital convolution.

1. Introduction

Most of the existing robot motion control methods include a trajectory generation method, in other words, they are very important for a trajectory generator as well as a tracking controller to achieve the robot motion control system with high (fast and accurate) performance. In many cases, the trajectory generation method affects a control system performance and its limitation. The generated trajectory over either rated speed (or max. velocity) or rated acceleration (or max. acceleration) cannot be well implemented just by a good tracking controller. Also, the trajectory generator should be realized with low computational burden for practical use. This is very core requirement in the trajectory generation method for real-time implementation.

Physically, since the jerk represents the change of acceleration with respect to time, abrupt jerk changes (abrupt changes of acceleration, ultimately, abrupt changes of force) is able to cause critical damages to the dynamic control systems and to induce unwanted vibrations or oscillations to the control system [1]. So, the jerk-bounded and smooth trajectory is required for preventing the control system from vibration/oscillation and mechanical wears of actuators. For high performance, the trajectory generator should satisfy the given system specifications such as maximum velocity, maximum acceleration and so on [2]. Actually, the control system will show bad performance if the generated trajectory is outside of the system specifications. Also, the trajectory generator must have a low computational complexity [3].

The remaining of this paper is organized as follows; section 2 introduces the conventional trajectory smoothing method based on digital convolution; section 3 suggests the jerk-bounded and smooth trajectory generation method satisfying the system requirements or specifications; section 4 shows the effectiveness of the suggested method through numerical simulations; finally, we provide the conclusion and future works.

*Corresponding author

2. Conventional Trajectory Generation using Convolution

The continuous-time and discrete-time convolutions have been used as the smooth trajectory generation method. For simplicity, let us focus on the one-dimensional (single axis) point-to-point (PTP) motion generation method with zero velocities as initial and final conditions. If the motion control system moves the given distance S with maximum velocity $v_{\max}$, then minimum movement time T_t can be expressed in the following form:

$$T_t = \frac{S}{v_{\max}} \tag{1}$$

where the area of left figure in Fig.1 implies the given distance S.

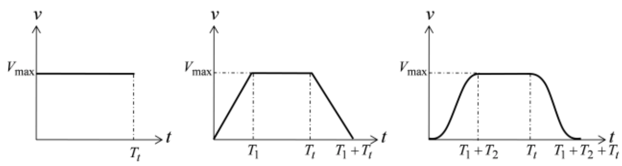

Fig. 1 Conventional smooth trajectory generation using convolution operation

Also, let us define the square function with a unit-area as following form:

$$H_i(t) = \begin{cases} \dfrac{1}{T_i}, & 0 \le t \le T_i \quad (i = 1, 2, \cdots) \\ 0, & otherwise \end{cases} \tag{2}$$

where $H_i(t)$ implies i-th square function. Here, if the left square function in Fig. 1 is convolved with the first square function $H_1(t)$ with T_1 satisfying $T_1 < T_t$, then we can get the trapezoidal trajectory of the mid figure in Fig. 1. Moreover, if the trapezoidal function of mid figure in Fig. 1 is convolved with the second square function $H_2(t)$ with T_2 satisfying $T_2 < T_1 < T_t$, then we can get the smooth trajectory of right figure in Fig. 1. Like this, we can see that twice convolutions bring the smooth trajectory as shown in Fig. 1. Also, all areas of three figures in Fig. 1 must be equal because the unit-area square functions are used for smoothing the trajectories. As the number of convolutions increase, the target trajectories must be more smoothed, however, movement time increases according to the increase of the number of convolutions as shown in Fig.1.

On the other hand, a motion planning method based on a digital convolution was proposed to reduce the computational burden in [4]. To begin with, let us make continuous-time i-th square function of Eq. (2) into the digital form $H_i(kT_s)$ with sampling time T_s. Surprisingly, if the digital input $X_i(kT_s)$ is convolved with $H_i(kT_s)$ in the discrete-time, then the output $Y_i(kT_s)$ is obtained as following recursive form:

$$Y_i(kT_s) = \frac{X_i(kT_s) - X_i((k-m_i)T_s)}{m_i} + Y_i((k-1)T_s) \quad (3)$$

where m_i is a positive integer satisfying $m_i = T_i / T_s$. This recursive form was proven in [4]. Also, we can know that one digital convolution requires just three additions and one multiplication. So, since the computational burden of digital convolution is very low, the trajectory generation method using digital convolution can be implemented in real-time.

3. Jerk-Bounded Trajectory Generation Method using Digital Convolution

To clarify the effect of convolution, let us redraw the mid figure in Fig. 1 obtained after applying one convolution as shown in Fig. 2. Here, we can find the maximum acceleration $a_{\max}$ defined as following form:

$$a_{\max} = \frac{v_{\max}}{T_1} \quad (4)$$

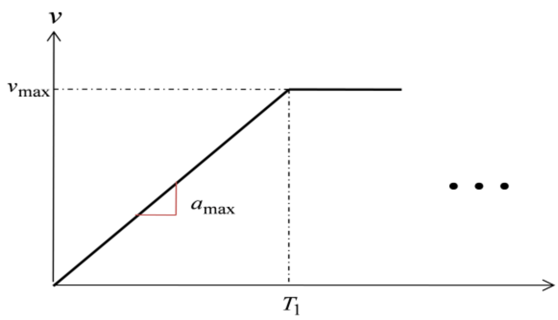

Fig. 2 Maximum acceleration of trapezoidal trajectory

If the convolution is applied one more to Eq. (4), then we can get the following relation between the maximum jerk and maximum acceleration:

$$j_{\max} = \frac{a_{\max}}{T_2} = \frac{v_{\max}}{T_1 T_2} \quad (5)$$

Without loss of generality, we can derive the following generalized relation from repetitive convolutions:

$$\therefore \quad v^{(i)}_{\max} = v_{\max} \prod_{j=1}^{i} \frac{1}{T_j} \quad (6)$$

where $v^{(i)}_{\max}$ denotes the i-times differentiation of $v_{\max}$, for example,

$$v^{(1)}_{\max} = a_{\max}$$
$$v^{(2)}_{\max} = j_{\max} \quad (7)$$

Actually, the motion control system has the specifications such as maximum (or rated) velocity, maximum (or rated) acceleration, and maximum jerk, in itself. These specifications become the design constraints for trajectory generation method to be suggested in the paper. For given specifications such as $v_{\max}$, $a_{\max}$, $j_{\max}$, firstly, we should find the time constraints T_1, T_2 from Eq. (4) and (5) satisfying $T_2 < T_1 < T_t$. Also, we should find a positive integer m_i satisfying $m_i = T_i / T_s$ to implement the recursive algorithm of Eq. (3). In the following section, we show the effectiveness of the suggested jerk-bounded trajectory generation method through numerical simulation.

4. Simulation and Conclusion

4.1 Simulation

Let us assume that the motion control system has the following specifications:

$$\begin{aligned} S &= 4[m] \\ v_{\max} &= 2[m/s] \\ a_{\max} &= 2[m/s^2] \\ j_{\max} &= 4[m/s^3] \\ T_s &= 0.001\,[s] \end{aligned} \quad (8)$$

Firstly, the time constraints satisfying $T_2 < T_1 < T_t$ are found as follows:

$$T_t = \frac{S}{v_{\max}} = 2[s]$$
$$T_1 = \frac{v_{\max}}{a_{\max}} = 1[s]$$
$$T_2 = \frac{a_{\max}}{j_{\max}} = 0.5[s]$$

Secondly, we can find positive integers m_1, m_2 as follows:

$$m_1 = T_1 / T_s = 1/0.001 = 1000$$
$$m_2 = T_2 / T_s = 0.5/0.001 = 500$$

Now, if the suggested recursive algorithm of Eq. (3) is applied to given specifications of Eq. (8), then we can get the simulation result as shown in Fig.3 after applying repetitive twice convolutions. As we can see in Fig. 3, the maximum jerk, acceleration and velocity satisfy the given specifications of Eq. (8). Moreover, if we would like to make jerk trajectory smooth as shown in Fig. 4, the more convolutions are required.

4.2 Conclusion and future works

This paper has proposed the jerk-bounded trajectory generation method using a digital convolution. The suggested jerk-bounded trajectory was able to reduce unexpected damages for the motion control system and to improve the tracking accuracy/speed of the control system. Also, it could be implemented in real-time because it requires just three additions and one multiplication per one convolution. Finally, the effectiveness of the suggested method was shown through numerical simulations for a point-to-point (PTP) motion generation application.

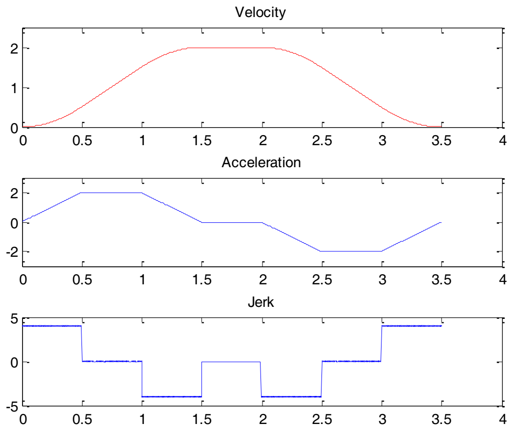

Fig. 3 Simulation result with twice convolutions

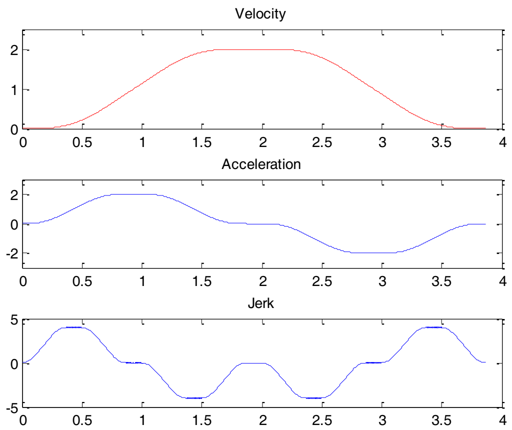

Fig. 4 Simulation result with four-times convolutions

Acknowledgement

This work was supported in part by a Korea Science and Engineering Foundation (KOSEF) grant funded by the Korea government (MEST) (R01-2008-000-20631), and in part by the Ministry of Knowledge Economy (MKE) and Korea Industrial Technology Foundation (KOTEF) through the Human Resource Training Project for Strategic Technology, and in part by Ministry of Knowledge Economy (MKE) under the Human Resources Development Program for Convergence Robot Specialists, and in part by Grant No.10032219 from Technology Innovation Program of the Ministry of Knowledge Economy, Republic of Korea.

References

[1] K. Ahn, W. K. Chung and Y. Youm, “Arbitrary States Polynomial-like Trajectory(ASPOT) Generation,” Proc. of IEEE Int. Conf. on Industrial Electronics Society, pp. 123-128, 2004.

[2] J. W. Jeon, and Y. Y. Ha, “A Generalized Approach for the Acceleration and Deceleration of Industrial Robots and CNC Machine Tools,” Proc. of IEEE Int. Conf. on Industrial Electronics Society, pp. 133-139, 2000.

[3] S. Macfarlane, and E. A. Croft, “Jerk-Bounded Manipulator Trajectory Planning: Design for Real-Time Applications,” Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 42-52, 2002.

[4] D. I. Kim, J. W. Jeon, and S. Kim, “Software Acceleration/Deceleration Methods for Industrial Robots and CNC Machining Tools” Proc. of Mechatronics, pp. 37-53, 1994.